PROTECTING DAISY CONTENT

Sebastien Hinderer[1,3], Dominique Burger[2,3], Bruno Marmol[4]

1 / Association BrailleNet 9
quai St Bernard
75005 Paris
FRANCE

2 / INSERM, UMR_S968
Institut de la Vision
75012 Paris
FRANCE

3 / UPMC Univ Paris 06, UMR_S968
Institut de la Vision
75012 Paris
FRANCE

4 / INRIA Grenoble-Rhone-Alpes
38334 Grenoble
FRANCE

ABSTRACT

DAISY has published a Specification for DAISY Protected Digital Talking Book. This paper discusses why such a specification is useful, not only for rightsholders but also for readers with print disabilities. An implementation of PDTB2 is proposed, called dtbprotect. It makes possible to simply produce an encrypted book from a book in DAISY format. It is currently experimented on the Helene Digital Library for the blind. It will be made available open source as to facilitate its implementation by other digital libraries.

1 Introduction

The development we present here has been carried out in the context of Helene, the Web service that BrailleNet created in 2002 to support the adaptation, the production and the secure delivery of books for people with print disabilities (BrailleNet 2009a / BrailleNet 2009b). The Helene project received the support of the French Ministry of Culture for it is based on a contractual approach with the publishing industry. Indeed, more that 100 publishers in France have signed contracts with BrailleNet for providing source files of their books (Desbuquois / Burger 2008). The main technical options chosen by BrailleNet for developing this service are:

- using XML DAISY standard ANSI NISO Z39.86 as a central format for storing and converting contents;
- using PKI standards for securing the delivery of these contents, as to comply with the requirements of both the French law and the publishers.

In 2007, the DAISY Consortium has published the DAISY Protected Digital Talking Book Specification (PDTB2) which provides a way to publish copyrighted DAISY material securely (DAISY Consortium 2007). Thus, PDTB2 was considered by BrailleNet as a potential solution in the future evolution of the Helene platform. However, surprisingly, although this specification is more than two years old, it is not yet widely used. Indeed, only a

few manufacturers provide devices that are able to read DAISY protected content, whereas protected books are delivered by only one content provider. One can wonder why a specification that seems interesting at first glance has received so little attention. Two explanations can be given:

- First, there may be some kind of reluctance in the community to the adoption of a highly secured format because it requires a good infrastructure and it limits the legitimate use that readers can do of books they receive.
- Second, protecting (or using protected) DAISY contents is a non-trivial task. Although there are some freely available software libraries that can be used, none of them does the entire job. There are no ready-to-use tools that make it possible to protect a Digital Talking Book.

This paper discusses these issues. In section 2, we discuss why the PDBT2 specification is of crucial importance not only for rightsholders but also for readers with print disabilities and for the future of the DAISY standard itself. In section 3, we introduce dtbprotect, a multi-platform tool that has been developed to protect any content conforming to the ANSI NISO Z39.86 2005-3 specification1.

2 Motivation for protecting DAISY books with PDTB2

Roughly speaking, PDTB2 specifies a way of encrypting DAISY books so that they can be decrypted only by authorised reading systems. Doing so is one of the necessary steps to ensure that content is delivered in accordance with the laws on intellectual property. The specification is thus satisfactory for rightsholders. However, as far as end users are concerned, the specification may become a problem because, while it prevents misuses of the content, it simultaneously results in limitation of legitimate uses. For instance, a protected book could be readable only on the device it has been authorised for, which may not offer all the reading facilities the user wishes. So the user may want to read the book on another device he is more comfortable with.

This problem may be one explanation why one part of the DAISY community seems to be reluctant to a wide use of the PDTB2 specification which has, though, been approved by the DAISY Consortium. This position could be summarized as follows: “Why should a specialised accessible format like DAISY be protected if this protection makes the reading of a book by end-users more difficult?”.

On the other hand, if a wider number of DAISY compliant reading devices were capable of reading protected contents, by implementing the PDTB2 specification as a standard, the burden on end-users would be lower.

Additionally, the more warranties one can offer to rightsholders with respect to intellectual property, the more they will accept to commit themselves to making their publications accessible. Given this principle and the cryptographic strength of the algorithms that have been chosen in PDTB2, this specification constitutes a strong point in favour of the DAISY community to negotiate with rightsholders. Consequently, a wider adoption of PDTB2 on DAISY compliant reading devices might foster the available books and be beneficial for the print disabled as well. Moreover, if one considers the perspective of promoting the DAISY standard itself as “the best way to read, the best way to publish“, in order to mainstream the production and distribution of adapted contents into the standard publishing chain, a security mechanisms complying with the DAISY format becomes a crucial necessity. Seen from this perspective, the PDTB2 specification can be considered as a significant step on the road leading to a much wider and less specific use of the DAISY format.

These observations and our work to continuously upgrade the Helene platform in compliance with the DAISY specification, led us to develop dtbprotect in order to facilitate the implementation of the PDTB2 specification.

1 referred as DAISY 3.0 in this paper

3 Dtbprotect – A tool for securing DAISY contents

Dtbprotect is a piece of software capable of producing a protected book from an unprotected DAISY 3.0 book and an encryption key as an input. This section provides details about the implementation of dtbprotect.

To give a few words of context, the tool has been developed to be used in the Helene digital library in order to prepare books and make them readable with PDTB2 capable platforms (BrailleNet 2009b). As Helene is a Web application, dtbprotect had to be invoked by scripts. Hence, it is a command-line oriented program whose integration in an automatic book generation process is easy: one just has to test its exit code to determine whether the program could complete its task successfully or not.

The second point worth mentioning is that the language used to develop this tool is Objective Caml. It is a functional programming language which is supported on a wide range of platforms including Microsoft Windows, Linux and Mac OS. Such languages are especially powerful to manipulate tree structures such as those found in XML documents, which is why the language has been chosen.

Third, the PDTB2 specification makes heavy use of two specifications published by the W3C: XML Encryption and XML Signature (W3C 2002 / W3C 2008). These specifications describe how encrypted and signed contents can be embodied in an XML document. They introduce elements and attributes to store not only the encrypted (signed) data, but also information about the encryption (signature) algorithms and keys.

As mentioned before, the PDTB2 specification makes a heavy use of these specifications. Indeed, a protected book is a file set that contains an authorisation object. This file gives information about the book: a reference to an unprotected content to be presented to the reader when the protected one cannot be decrypted, the list of protected files, the corresponding rights, the keys used to encrypt files, each key being itself encrypted with the public key of the target reading system. So, the authorisation object is one place where XML encryption and XML signature are used. The XML files of the book, such as SMIL files and the text itself may be other places where XML encryption is used.

For implementing the XML encryption and XML signature we used a tool called xmlsec, a free software based on the well known libxml2 XML library, which is both a library and a command-line tool (XML soft 2009). Using it as a command-line tool was not possible in our case, because it is not flexible enough. More precisely, the problem was that one wants to generate a key which has to be used to encrypt the files to protect and then stored encrypted in the authorisation object. In particular, the key would have to be stored unencrypted on disk, which is something one may want to avoid, for security reasons.

All these considerations have led to the dtbprotect tool as it is now: an executable program that uses the xmlsec library and can be called from the command-line. As an example, assuming one wants to encrypt a book whose package file is located in /tmp/book.opf with an encryption key located in /tmp/key.asc with the protected book being stored in the /tmp/output directory, one would use the following command-line:

```
$ dtbprotect --input /tmp/book.opf --key /tmp/key.asc --output /tmp/output/
```

If the program succeeds then the /tmp/output/ directory will contain a protected file set which is almost ready to be presented to a reading system. The only missing part is the facade (that is, the unprotected content used as a fallback if reading the protected files is for some reason not possible). Since our DAISY file sets are produced from XML DTBook files, we produce the facade this way. However, in the general case, one may produce the facade another way, so we will probably have to wait to see what people need to figure out whether facade generation can be integrated to dtbprotect or not and how.

One final remark is due. The PDTB2 specification does not mention any limitation of any kind regarding the size of files that can be used, etc. However, in practice some DAISY readers are portable devices with limited

resources, which means they are not always able to process arbitrarily large files. Thus, it may be suitable for an implementation to be able to accept some parameters to limit the size of produced files. That being said, estimating the size of output files given the sizes of input files is hard, because the modification of size implied by the transformations is not always well known a priori.

4 Conclusion

Dtbprotect has been developed and implemented by BrailleNet on the Helene platform. It makes possible to produce secured DAISY contents in compliance with the PDTB2 specification of DAISY. This tool is available open source in order to make as smooth as possible the implementation of copyright control mechanisms. We are currently discussing with manufacturers the possibility for them to implement this specification so that thousands of books of the Server Helene could be available to end-users worldwide on various reading devices.

5 Acknowledgments

We are grateful to the French Ministry of Culture and Communication and the company Alcatel-Lucent for their support and cooperation.

References

BrailleNet (2009a). Helene server-Bibliothèque helénè. Retrieved from: http://www.serveur-helene.org

BrailleNet (2009b).Bibliothèque Hélène: http://www.bibliotheque-helene.org

Desbuquois C. / Burger D. (2008). Lessons from the Hélène Digital Library. In: 2nd European eAccessibility Forum-Accessible e-books: An opportunity for the disabled. Retrieved from: http://inova.snv.jussieu.fr/evenements/colloques/colloques/article.php?c=46&l=en&a=81#contenu_article

DAISY Consortium (2007). Specification for DAISY Protected Digital Talking Book (PDTB). Retrieved from: http://www.daisy.org/projects/pdtb/daisy-pdtb-spec.html

W3C (2002). XML Encryption Syntax and Processing. Retrieved from: http://www.w3.org/TR/xmlenc-core/

W3C (2008). XML Signature Syntax and Processing (Second Edition).Retrieved from: http://www.w3.org/TR/xmldsig-core/

XML soft (2009). XML Library. Retrieved from: http://xmlsoft.org/